ETH zürich
Departement Informatik

Theoretische Informatik

Prof. Dr. Juraj Hromkovič
Dr. Hans-Joachim Böckenhauer
`http://www.ita.inf.ethz.ch/theoInf18`

Lösungsvorschläge – Blatt 1

Zürich, 28. September 2018

Lösung zu Aufgabe 1

(a) Sei $w = a_1 a_2 \dots a_m$ ein Wort mit $m \leq n$ verschiedenen Buchstaben. Dann sind offensichtlich alle möglichen Teilwörter paarweise verschieden. Es bleibt also nur die Anzahl der Teilwörter von w zu zählen. Für ein Teilwort der Länge $i \geq 1$ gibt es $m - i + 1$ mögliche Anfangspositionen in w, denn die Positionen $m - i + 2, \dots, m$ sind nicht möglich. Aufsummieren über alle möglichen Längen nichtleerer Teilwörter ergibt insgesamt

$$\sum_{i=1}^{m}(m - i + 1) = m \cdot (m+1) - \sum_{i=1}^{m} i = m \cdot (m+1) - \frac{m \cdot (m+1)}{2} = \frac{m \cdot (m+1)}{2}$$

nichtleere Teilwörter. Dazu kommt noch das leere Wort, das Teilwort von jedem Wort ist, insgesamt gibt es also

$$1 + \frac{m \cdot (m+1)}{2}$$

verschiedene Teilwörter von w.

(b) Wir zählen zunächst, wie viele Wörter der Länge n über $\{a, b, c\}$ die Bedingung aus der Aufgabenstellung *nicht* erfüllen. Dies sind alle diejenigen Wörter, die höchstens zwei der Buchstaben enthalten. Es gibt 2^n verschiedene Wörter, die nur die Buchstaben a und b enthalten, denn für jede der n Positionen kann einer der beiden Buchstaben gewählt werden. Analog gibt es 2^n verschiedene Wörter, die nur a und c enthalten, sowie 2^n verschiedene Wörter, die nur b und c enthalten. Unter diesen $3 \cdot 2^n$ Wörtern haben wir die Wörter a^n, b^n und c^n jeweils doppelt gezählt, also gibt es insgesamt $3 \cdot 2^n - 3$ Wörter, die die Bedingung der Aufgabenstellung nicht erfüllen. Weil es insgesamt 3^n Wörter der Länge n über $\{a, b, c\}$ gibt, haben wir insgesamt also

$$3^n - 3 \cdot 2^n + 3$$

Wörter, die jeden der drei Buchstaben mindestens einmal enthalten.

(bitte wenden)

Lösung zu Aufgabe 2

(a) Sei Σ ein Alphabet und seien $u, v \in \Sigma^*$. Dann existieren $k, l \in \mathbb{N}$ mit $u = u_1 u_2 \dots u_k$ und $v = v_1 v_2 \dots v_l$, wobei $u_1, \dots, u_k, v_1, \dots, v_l \in \Sigma$. Damit ist

$$uv = u_1 u_2 \dots u_k v_1 v_2 \dots v_l$$

und somit

$$\begin{aligned}(uv)^{\mathsf{R}} &= v_l \dots v_2 v_1 u_k \dots u_2 u_1 \\ &= v^{\mathsf{R}} u^{\mathsf{R}}\,,\end{aligned}$$

weil $v^{\mathsf{R}} = v_l \dots v_2 v_1$ und $u^{\mathsf{R}} = u_k \dots u_2 u_1$ gilt.

(b) Wir wählen

$$\begin{aligned}L_1 &= \{\lambda, 0\}\,, \\ L_2 &= \{0^{2i} \mid i \in \mathbb{N}\}\,, \\ L_3 &= \{\lambda\} \cup \{0^{2i+1} \mid i \in \mathbb{N}\}\,.\end{aligned}$$

Dann gilt $L_2 \cap L_3 = \{\lambda\}$ und somit ist

$$L_1 \cdot (L_2 \cap L_3) = \{\lambda, 0\}$$

eine endliche Sprache. Offenbar gilt aber auch

$$L_1 \cdot L_2 = \{0^i \mid i \in \mathbb{N}\}$$

und

$$L_1 \cdot L_3 = \{\lambda, 0\} \cup \{0^i \mid i \in \mathbb{N} - \{0\}\} = \{0^i \mid i \in \mathbb{N}\}$$

Somit

$$L_1 \cdot L_2 \,\cap\, L_1 \cdot L_3 = \{0^i \mid i \in \mathbb{N}\} = \Sigma^*$$

unendlich.

Lösung zu Aufgabe 3

(a) Wir wählen $\Sigma = \{a\}$ und

$$L_1 = \{a, aa, \dots, a^k\} = \{a^i \mid 1 \leq i \leq k\}\,.$$

Dann gilt für

$$L_2 = \{\lambda, a\}\,,$$

dass

$$\begin{aligned}L_1 \cdot L_2 &= L_1 \cdot \{\lambda\} \cup L_1 \cdot \{a\} \\ &= L_1 \cup \{a^i a \mid 1 \leq i \leq k\} \\ &= L_1 \cup \{a^i \mid 2 \leq i \leq k+1\} \\ &= \{a^i \mid 1 \leq i \leq k+1\}\,,\end{aligned}$$

also ist $|L_1 L_2| = k + 1$.

(b) Hier lässt sich ein Beispiel analog zu Aufgabenteil (a) konstruieren. Wir wählen wieder $\Sigma = \{a\}$ und

$$L_1 = \{a, aa, \ldots, a^k\} = \{a^i \mid 1 \leq i \leq k\}\,.$$

Dann gilt für

$$L_2 = \{\lambda, a, aa, aaa, aaaa, aaaaa\}\,,$$

dass

$$\begin{aligned} L_1 \cdot L_2 &= \bigcup_{j=0}^{5} L_1 \cdot \{a^j\} \\ &= \bigcup_{j=0}^{5} \{a^i a^j \mid 1 \leq i \leq k\} \\ &= \{a^i \mid 1 \leq i \leq k+5\}\,, \end{aligned}$$

also ist $|L_1 L_2| = k + 5$.

(c) Wir wählen $\Sigma = \{a, b\}$ und

$$L_1 = \{a^i \mid 1 \leq i \leq k\}$$

und

$$L_2 = \{b^i \mid 1 \leq i \leq l\}$$

Dann ist

$$L_1 L_2 = \{a^i b^j \mid 1 \leq i \leq k \text{ und } 1 \leq j \leq l\}\,,$$

also $|L_1 L_2| = k \cdot l$.

(d) Um dieselbe Grösse von $L_1 \cdot L_2$ über einem einbuchstabigen Alphabet zu erreichen, müssen wir sicherstellen, dass jedes Paar von Wörtern aus L_1 und L_2 ein unterschiedliches Gesamtwort erzeugt. Dies können wir erreichen, wenn wir die Längen der Wörter in L_1 so wählen, dass wir für jedes $v \in L_1$ durch das Anhängen eines beliebigen Wortes $w \in L_2$ nicht das nächstlängere Wort aus L_1 erreichen können. Die Idee ist also, die Längen der Wörter in L_2 möglichst klein zu halten und die Längen der Wörter in L_1 hinreichend unterschiedlich zu wählen. Sei $\Sigma = \{a\}$. Dann setzen wir

$$L_1 = \{a^{li} \mid 0 \leq i \leq k-1\}$$

und

$$L_2 = \{a^j \mid 0 \leq j \leq l-1\}\,.$$

Dann ist

$$L_1 L_2 = \{a^{li} a^j \mid 0 \leq i \leq k-1 \text{ und } 0 \leq j \leq l-1\}\,.$$

Alle Wörter in $L_1 L_2$ haben paarweise unterschiedliche Längen, weil alle Wortlängen in L_1 Vielfache von l sind und alle Wörter in L_2 kürzer als l sind. Also gilt $|L_1 L_2| = k \cdot l$.